APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/817,437	03/26/2001	Stephen J. Todd	JW-EMC-007	9135

		EXAMINER
7590	06/07/2004	RAMPURIA, SATISH

Robert L. Dulaney
EMC Corporation
35 PARKWOOD DRIVE
HOPKINTON, MA 01748

ART UNIT	PAPER NUMBER
2124	

DATE MAILED: 06/07/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

-- *The MAILING DATE of this communication appears on the cover sheet with the correspondence address* --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>03</u> MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed
  after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any
  earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on <u>*26 March 2001*</u>.

2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is
closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-89* is/are pending in the application.

   4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-89* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

   Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

   Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

   a)☐ All   b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage

      application from the International Bureau (PCT Rule 17.2(a)).

   * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☒ Notice of References Cited (PTO-892)
2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3)☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
   Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413)
   Paper No(s)/Mail Date. _____ .
5)☐ Notice of Informal Patent Application (PTO-152)
6)☐ Other: _____.

DETAILED ACTION

1.   This action is in response to the application filed on 03/26/2001.

2.   Claims 1-89 are pending.

Specification

3.   The disclosure is objected to because it contains an embedded hyperlink and/or other

form of browser-executable code. On page 7, lines 22 and 23. Applicant is required to

delete the embedded hyperlink and/or other form of browser-executable code. See MPEP

§ 608.01.

4.   The disclosure is objected to because of the following informalities:

On page 3, line 9 "!" should be "." and page 4, line 22 the word "extracting banking
information" missing a (close quote) """".

Appropriate correction is required

Information Disclosure Statement

5.   Applicant's IDS form 1449, has not been provided, therefore, references (Common

Information Model (CIM) Specification" Version 2.2, dated Jun. 14, 1999 prepared by

the Distributed Management Task Force, Inc. (DMTF) and "Specification for CIM

Operations of HTTP" Version 1.0, dated Aug. 11, 1999, prepared by DMTF, disclosed on

page 7 and 8, lines18-20 and 1-2, respectively) used in the application has been included

in PTO 892.

Drawings

6.   Formal drawings are required in response to the instant Office action.

7.   The following informality has been noted and requires correction in response to this

Office Action. Figure 2 (A-B) should be designated by a legend such as --Prior Art--

because only that which is old is illustrated. See MPEP § 608.02(g).

A proposed drawing correction or corrected drawings are required in reply to the Office

action to avoid abandonment of the application. The objection to the drawings will not be held

in abeyance.

Claim Rejections - 35 USC § 112, second paragraph

8.   Claims 11, 23, 37, 51, 52, 74, and 88 are rejected under 35 U.S.C. 112, second paragraph,

as being indefinite for failing to particularly point out and distinctly claim the subject

matter which applicant regards as the invention.

Clarification and/or correction are required.

Regarding, claim 11, on line 14, the limitation, "other-component" is unclear as to what other

component management software is selected from.

Regarding, claim 37, on lines 11 and 21, the limitation, "other architecture" is unclear as to

what other architecture is not compatible with language.

Regarding, claim 52, on line 11, the limitation, "other computer devices" is unclear as to what

other computer devices are operating under second architecture. And on line 16, the limitation,

"other particular information" is unclear as to what other particular information related to the

result.

Claim 23, has the similar limitation to those in claim 11 with respect to "other-component",

recited on the line 9.

Claim 51, has the similar limitation to those in claim 11 with respect to "other-component",

recited on the line 3.

Claim 74, has the similar limitation to those in claim 11 with respect to "other-component",

recited on the line 20.

Claim 88, has the similar limitation to those in claim 11 with respect to "other-component",

recited on the line 18.

The rejection of the base claim is necessarily incorporated into the dependent claims.

Claim Rejections - 35 USC § 103

9.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

10.   Claims 1-62 and 75-89 are rejected under 35 U.S.C. 103(a) as being unpatentable over

US Patent No. 6,086,622 to Abe et al., hereinafter called Abe, in view of admitted prior

art and further in view of US Patent No. 5,295,256 to Bapat hereinafter called Bapat.

Per claims 1, 4, 5, 6, and 9:

Abe discloses:

- A computer system employing management software written in a first computer language compatible with first architecture and not compatible with second architecture (col. 1,

           lines 9-14 "a method and an apparatus for converting a program for a computer of a first architecture to a machine program adapted for a computer of a second architecture").

- means for converting said called public functions and/or data attributes (col. 1, lines 62-63 "convert a program (functions, attributes, data etc.) for a computer of a first architecture") to representations of said called public functions and/or data attributes formed in a different computer language compatible with said second architecture (col. 1, lines 63-64 "to a program (functions, attributes, data etc.) for a computer of a second architecture").

Abe does not explicitly disclose a schema formed within said first architecture; header files contained within said schema, said header files being represented in said first language and capable of being utilized by said management software.

However, admitted prior art discloses in an analogous computer system a schema formed within said first architecture (Applicant's specification, page 4, lines 2-3 "These architectures are combinations of software such as schemas, languages and protocols"); header files contained within said schema, said header files being represented in said first language and capable of being utilized by said management software (Applicant's specification, page 4, lines 7-8 "And on top of that computer language is a schema (header-related software...) such as that derived from or implemented in RAID++. RAID++ is an object-oriented representation of a CLARiiON®").

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method of including the header file within the schema of an architecture as taught admitted prior art in corresponding to method of converting a program of a first architecture to a second architecture as taught by Abe. The modification would be obvious because of one of ordinary skill in the art would be motivated to include prior art header files within schema of an architecture to provide less arduous code generation to

improve the communication within the corporate as suggested in admitted prior art (Applicant's specification, pages 2 and 3, lines 23-25 and 1-20, respectively).

Neither Abe nor admitted prior art explicitly disclose means for manipulating said header files to locate public functions and/or data attributes of said header files; means, responsive to operation of said manipulating means, for emitting code that calls said public functions and/or data attributes in said first language to obtain called public functions and/or data attributes.

However, Bapat discloses in an analogous computer system means for manipulating (col. 3, lines 53-54 "manipulating a representation of the object class... stored... table schema") said header files (col. 9, lines 1-2 "C++ source and header files are opened") to locate public functions and/or data attributes of said header files (col. 6, lines 28-29 "it is desirable to be able to provide... storage of the attributes of these objects managed by the network management system"); means, responsive to operation of said manipulating means(col. 3, lines 53-54 "manipulating a representation of the object class... stored... table schema"), for emitting code that calls said public functions and/or data attributes in said first language to obtain called public functions and/or data attributes (col. 13, lines 15-16 "Control... passes to... attribute insertion routine is called of each attribute").

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method of manipulating header files as taught by Bapat into the method of converting a program of a first architecture to a second architecture as taught by the combination system of Abe and admitted prior art. The modification would be

obvious because of one of ordinary skill in the art would be manipulate the header files for

mapping the within the schema of a database as suggested by Bapat (col. 2, lines 6-17).

Per claim 2:

Neither Abe nor Bapat explicitly disclose forwarding said representations to desired destinations

within and beyond said system.

However, admitted prior art discloses in an analogous computer system forwarding said

representations to desired destinations within and beyond said system (Applicant's

specification, pages 5, lines 16-17 "both defining itself and its relationships to other objects in

the computer system or network. An object can send and receive messages to and from other

objects").

Therefore, it would have been obvious to a person of ordinary skill in the art at the time

the invention was made to incorporate the method of send / receive message to another object

within system or to the network as taught admitted prior art in corresponding to method of

converting a program of a first architecture to a second architecture as taught in the combination

system by Abe and Bapat. The modification would be obvious because of one of ordinary skill in

the art would be motivated to send / receive message over the network or system when one needs

to get fetch the data as suggested in admitted prior art (Applicant's specification, pages 2 and 3,

lines 23-25 and 1-20, respectively).

Per claims 3, and 7:

Neither Abe nor Bapat explicitly disclose said first computer language is RAID++ and said different computer language is XML/CIM.

However, admitted prior art discloses in an analogous computer system said first computer language is RAID++ (Applicant's specification, pages 4, lines 6-9 "Object-oriented computer language C++... a schema... that derived from or implemented in RAID++") and said different computer language is XML/CIM (Applicant's specification, pages 7, lines 12-13 "the aforementioned XML computer language; and, on top of XML is a new and advantageous schema called Common Information Model (CIM)").

The feature of using the languages would be obvious for the reasons set forth in the rejection of claim 1.

Per claim 8:

Neither Abe nor Bapat explicitly disclose said first computer language is a first object-oriented language capable of pictorial representation typically in a parent-child tree configuration and said different computer language is a second object-oriented language capable of pictorial representation typically in a flat database configuration.

However, admitted prior art discloses in an analogous computer system first computer language is a first object-oriented language capable of pictorial representation typically in a parent-child tree configuration (Applicant's specification, pages 6, lines 2-6 "C++ computer language... the relationship between these specific objects in the storage processor is usually visualized or characterized as a "tree" of objects") and said different computer language is a second object-oriented language capable of pictorial representation typically in a flat database

configuration (Applicant's specification, pages 6, lines 17-22 "standard that avoids object trees and arranges all objects in a database where a "flat" relationship is obtained").

The feature of using the object-oriented languages would be obvious for the reasons set forth in the rejection of claim 1.

Per claim 10:

Neither Abe nor Bapat explicitly disclose said management software is storage management software.

However, admitted prior art discloses in an analogous computer system management software is storage management software (Applicant's specification, pages 7, lines 5-6 "Web technologies to manage enterprise systems such as storage systems").

The feature of storage management software would be obvious for the reasons set forth in the rejection of claim 1.

Per claim 11:

Neither Abe nor Bapat explicitly disclose said management software is selected from the group consisting of storage, printer, server and other-component management software.

However, admitted prior art discloses in an analogous computer system management software is selected from the group consisting of storage, printer, server and other-component management software (Applicant's specification, pages 2, lines 15-17 "software which runs on and controls that hardware such as operating systems software and applications software such as peripheral-device management software")

The feature of storage management software would be obvious for the reasons set forth in the rejection of claim 1.

Per claims 12, 52, 57, 58, 59, 62, 75, 76, 81, and 89:

Abe discloses:

- A computer network employing a computer system utilizing management software written in a first computer language compatible with first architecture and not compatible with second architecture (col. 1, lines 9-14 "a method and an apparatus for converting a program for a computer of a first architecture to a machine program adapted for a computer of a second architecture").
- converts said called public functions and/or data attributes to representations of said called public functions and/or data attributes formed in a different computer language compatible with said second architecture (col. 1, lines 62-63 "convert a program (functions, attributes, data etc.) for a computer of a first architecture" and col. 1, lines 63-64 "to a program (functions, attributes, data etc.) for a computer of a second architecture").

Abe does not explicitly disclose a schema formed within said first architecture; header files contained within said schema, said header files being represented in said first language and capable of being utilized by said management software.

However, admitted prior art discloses in an analogous computer system a schema formed within said first architecture (Applicant's specification, page 4, lines 2-3 "These architectures are combinations of software such as schemas, languages and protocols"); header files contained within said schema, said header files being represented in said first language and capable of being utilized by said management software (Applicant's specification, page 4, lines 7-8 "And on top of that computer language is a schema (header-related software...) such as that derived from or implemented in RAID++. RAID++ is an object-oriented representation of a CLARiiON®").

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method of including the header file within the schema of an architecture as taught admitted prior art in corresponding to method of converting a program of a first architecture to a second architecture as taught by Abe. The modification would be obvious because of one of ordinary skill in the art would be motivated to include prior art header files within schema of an architecture to provide less arduous code generation to improve the communication within the corporate as suggested in admitted prior art (Applicant's specification, pages 2 and 3, lines 23-25 and 1-20, respectively).

Neither Abe nor admitted prior art explicitly disclose apparatus for manipulating said header files to locate public functions and/or data attributes of said header files; and, apparatus, responsive to operation of said manipulating apparatus, for emitting code that calls said public functions and/or data attributes in said first language to obtain called public functions and/or data attributes.

However, Bapat discloses in an analogous computer system apparatus for manipulating (col. 3, lines 53-54 "manipulating a representation of the object class... stored... table schema") said header files (col. 9, lines 1-2 "C++ source and header files are opened") to locate public functions and/or data attributes of said header files (col. 6, lines 28-29 "it is desirable to be able to provide... storage of the attributes of these objects managed by the network management system"); apparatus, responsive to operation of said manipulating means(col. 3, lines 53-54 "manipulating a representation of the object class... stored... table schema"), for emitting code that calls said public functions and/or data attributes in said first language to obtain called

public functions and/or data attributes (col. 13, lines 15-16 "Control... passes to... attribute insertion routine is called of each attribute").

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method of manipulating header files as taught by Bapat into the method of converting a program of a first architecture to a second architecture as taught by the combination system of Abe and admitted prior art. The modification would be obvious because of one of ordinary skill in the art would be manipulate the header files for mapping the within the schema of a database as suggested by Bapat (col. 2, lines 6-17).

Claim 13 is the apparatus claim corresponding to system claim 2 and rejected under the same rational set forth in connection with the rejection of claim 2 above.

Per claims 14, 18, 79, 80, and 84:

Neither Abe nor Bapat explicitly disclose said first computer language is RAID++ and said different computer language is XML/CIM.

However, admitted prior art discloses in an analogous computer system said first computer language is RAID++ (Applicant's specification, pages 4, lines 6-9 "Object-oriented computer language C++... a schema... that derived from or implemented in RAID++") and said different computer language is XML/CIM (Applicant's specification, pages 7, lines 12-13 "the aforementioned XML computer language; and, on top of XML is a new and advantageous schema called Common Information Model (CIM)").

The feature of using the languages would be obvious for the reasons set forth in the rejection of claim 12.

Claim 15 is the apparatus claim corresponding to system claim 4 and rejected under the same rational set forth in connection with the rejection of claim 4 above.

Claims 16, 20 are the apparatus claim corresponding to system claim 5 and rejected under the same rational set forth in connection with the rejection of claim 5 above.

Claim 17 is the apparatus claim corresponding to system claim 6 and rejected under the same rational set forth in connection with the rejection of claim 6 above.

Per claims 19, 78, and 83:

Neither Abe nor Bapat explicitly disclose said first computer language is a first object-oriented language capable of pictorial representation typically in a parent-child tree configuration and said different computer language is a second object-oriented language capable of pictorial representation typically in a flat database configuration.

However, admitted prior art discloses in an analogous computer system first computer language is a first object-oriented language capable of pictorial representation typically in a parent-child tree configuration (Applicant's specification, pages 6, lines 2-6 "C++ computer language... the relationship between these specific objects in the storage processor is usually visualized or characterized as a "tree" of objects") and said different computer language is a second object-oriented language capable of pictorial representation typically in a flat database configuration (Applicant's specification, pages 6, lines 17-22 "standard that avoids object trees and arranges all objects in a database where a "flat" relationship is obtained").

The feature of using the object-oriented languages would be obvious for the reasons set forth in the rejection of claim 12.

Per claims 21, 54, 61, 77, and 87:

The rejection of claim 12 is incorporated, and further, Abe disclose:

- a SAN which communicates with and is controlled by said computer system (col. 5, lines 1-5 "fetched instruction storing section…stores… instructions… fetch section…").

Per claims 22, 53, 60, 85, and 86:

Neither Abe nor Bapat explicitly disclose said management software is storage management software.

However, admitted prior art discloses in an analogous computer system management software is storage management software (Applicant's specification, pages 7, lines 5-6 "Web technologies to manage enterprise systems such as storage systems").

The feature of storage management software would be obvious for the reasons set forth in the rejection of claim 12.

Per claims 23, 56, and 88:

Neither Abe nor Bapat explicitly disclose said management software is selected from the group consisting of storage, printer, server and other-component management software.

However, admitted prior art discloses in an analogous computer system management software is selected from the group consisting of storage, printer, server and other-component

management software (Applicant's specification, pages 2, lines 15-17 "software which runs on

and controls that hardware such as operating systems software and applications software such as

peripheral-device management software")

The feature of storage management software would be obvious for the reasons set forth in

the rejection of claim 12.

Claims 24-31 are the method claims corresponding to system claims 1-8, respectively, and

rejected under the same rational set forth in connection with the rejection of claims 1-8,

respectively, above.

Claims 32 and 33 are the method claims corresponding to system claim 5 and rejected under the

same rational set forth in connection with the rejection of claim 5 above.

Per claims 34 and 82:

The rejection of claim 24 is incorporated, and further, Abe disclose:

- said first architecture is legacy architecture and said second architecture is non-legacy

   architecture (col. 3, lines 1-11 "... machine program... depends on the computer...

   second architecture... converted... high-level language source program (which does not

   depend on any architecture)... programs for computers of different architectures... be

   produced easily, or a program for a computer of the first architecture can be easily

   converted to a program for a computer of another (the second) architecture").

Claims 35 and 36 are the method claims corresponding to system claims 10-11, respectively, and

rejected under the same rational set forth in connection with the rejection of claims 10-11,

respectively, above.

Claims 37 and 38 are the computer program product claim corresponding to system claims 1

and 2, respectively, and rejected under the same rational set forth in connection with the rejection

of claims 1 and 2, respectively, above.

Claim 39 is the computer program product claim corresponding to system claim 4 and rejected

under the same rational set forth in connection with the rejection of claim 4 above.

Claim 40 is the computer program product claim corresponding to system claim 3 and rejected

under the same rational set forth in connection with the rejection of claim 3 above.

Claims 41-44 are the computer program product claim corresponding to system claims 5-8,

respectively, and rejected under the same rational set forth in connection with the rejection of

claims 5-8, respectively, above.

Claim 45 is the computer program product claim corresponding to system claim 5 and rejected

under the same rational set forth in connection with the rejection of claim 5 above.

Claim 47 is the computer program product claim corresponding to system claim 10 and rejected

under the same rational set forth in connection with the rejection of claim 10 above.

Claim 48 is the computer program product claim corresponding to system claim 11 and rejected

under the same rational set forth in connection with the rejection of claim 11 above.

Claim 49 is the computer program product claim corresponding to system claim 1 and rejected

under the same rational set forth in connection with the rejection of claim 1 above.

Claim 50 is the computer program product claim corresponding to system claim 10 and rejected

under the same rational set forth in connection with the rejection of claim 10 above.

Claim 51 is the computer program product claim corresponding to system claim 11 and rejected

under the same rational set forth in connection with the rejection of claim 11 above.

Per claim 55:

The rejection of claim 52 is incorporated, and further, Abe disclose:

- said first architecture is legacy architecture and said second architecture is non-legacy
architecture (col. 3, lines 1-11 "... machine program... depends on the computer... second
architecture... converted... high-level language source program (which does not depend on any
architecture)... programs for computers of different architectures... be produced easily, or a
program for a computer of the first architecture can be easily converted to a program for a
computer of another (the second) architecture").

11.   Claims 63-74 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent

No. 6,086,622 to Abe et al., hereinafter called Abe, in view of admitted prior art.

Per claim 63 and 67:

Abe discloses:

- In a computer system compatible with computer architecture, management software (col.
1, lines 9-14 "a method and an apparatus for converting a program for a computer of a
first architecture to a machine program adapted for a computer of a second
architecture").
- software means for converting said responses to equivalent responses compatible with
said first language and for communicating said equivalent responses (col. 1, lines 62-63
"convert a program (functions, attributes, data etc.) for a computer of a first
architecture") to the destination from which, or to destinations related to that from
which, said first requests originated (col. 1, lines 63-64 "to a program (functions,
attributes, data etc.) for a computer of a second architecture").

Abe does not explicitly disclose software means for obtaining responses to said first requests in second language compatible with said computer architecture.

However, admitted prior art discloses in an analogous computer system software means for obtaining responses to said first requests in second language compatible with said computer architecture (Applicant's specification, page 4, lines 2-3 "These architectures are combinations of software such as schemas, languages and protocols"); header files contained within said schema, said header files being represented in said first language and capable of being utilized by said management software (Applicant's specification, page 4, lines 7-8 "And on top of that computer language is a schema (header-related software…) such as that derived from or implemented in RAID++. RAID++ is an object-oriented representation of a CLARiiON®"").

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method of including the header file within the schema of an architecture as taught admitted prior art in corresponding to method of converting a program of a first architecture to a second architecture as taught by Abe. The modification would be obvious because of one of ordinary skill in the art would be motivated to include prior art header files within schema of an architecture to provide less arduous code generation to improve the communication within the corporate as suggested in admitted prior art (Applicant's specification, pages 2 and 3, lines 23-25 and 1-20, respectively).

Per claim 64:

The rejection of claim 63 is incorporated, and further, Abe disclose:

-   said first architecture is legacy architecture and said second architecture is non-legacy

    architecture (col. 3, lines 1-11 "… machine program… depends on the computer…

    second architecture… converted… high-level language source program (which does not

    depend on any architecture)… programs for computers of different architectures… be

    produced easily, or a program for a computer of the first architecture can be easily

    converted to a program for a computer of another (the second) architecture").

Per claim 65:

The rejection of claim 64 is incorporated, and further, Abe does not explicitly disclose said

management software is storage management software.

However, admitted prior art discloses in an analogous computer system management

software is storage management software (Applicant's specification, pages 7, lines 5-6 "Web

technologies to manage enterprise systems such as storage systems").

The feature of storage management software would be obvious for the reasons set forth in

the rejection of claim 63.

Per claim 66:

The rejection of claim 64 is incorporated, and further, Abe disclose:

-   a SAN which communicates with and is controlled by said computer system (col. 5, lines

    1-5 "fetched instruction storing section…stores… instructions… fetch section…").

Claim 68 is the computer program product claim corresponding to system claim 63 and rejected

under the same rational set forth in connection with the rejection of claim 63 above.

Claim 69 is the computer program product claim corresponding to system claim 64 and rejected

under the same rational set forth in connection with the rejection of claim 64 above.

Claim 70 is the method claim corresponding to system claim 63 and rejected under the same

rational set forth in connection with the rejection of claim 63 above.

Claim 71 is the method claim corresponding to system claim 64 and rejected under the same

rational set forth in connection with the rejection of claim 64 above.

Claim 72 is the method claim corresponding to system claim 62 and rejected under the same

rational set forth in connection with the rejection of claim 62 above.

Claim 73, is the method claim corresponding to system claim 21 and rejected under the same

rational set forth in connection with the rejection of claim 21 above.

Claim 74 is the method claim corresponding to system claim 11 and rejected under the same

rational set forth in connection with the rejection of claim 11 above.

Conclusion

12.   The prior art made of record and not relied upon is considered pertinent to applicant's

disclosure.

The following patent is cited to further show the state of the art with respect to converting

management software.

US Pub. No. 2002/0066085 to Nowlin Jr. et al.

US Patent No. 5,551,015 to Goettelmann et al.

US Patent No. 5,307,504 to Robinson et al.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Satish S. Rampuria whose telephone number is 703-305-8891. The examiner can normally be reached on 8:30 am to 5:00 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kakali Chaki can be reached on (703) 305-9662. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Satish S. Rampuria
Patent Examiner
Art Unit 2124
06/07/2004

KAKALI CHAKI
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100